REMARKS

Examiner has objected to the Specification because of various informalities. Applicant has amended the Specification to correct the informalities pointed out by Examiner.

Examiner has rejected claims 1 and 2 under 35 U.S.C. § 102(e) as being anticipated by USPN 6,108,738 (Chambers). Applicant respectfully traverses the rejection.

Applicant has amended claim 1 to make it clear that the internal bus and the logic block are all located within a single integrated circuit. This alone clearly distinguishes the interface block set out in claim 1 from bridge 480 discussed by Chambers.

In Chambers, bridge 480, shown in Figure 5, couples an internal PCI bus 416 to an external PCI bus 490. See column 9, lines 19 through 20. External PCI bus 490 is clearly not a socket of a logic block located on the same integrated circuit as bridge 480.

Further, the elements of claim 1 are four separate modules: a synchronization module, a translation module, a queue module and a driver module. Chambers does not disclose or suggest the use of separate modules performing the functionality set out by claim 1 of the present case.

When setting out the rationale for the rejection of claims 1 and 2, Examiner points to locations in Chambers where functionality is described. For example, Examiner argue that the synchronization module, the translation module and the queue module set out in claim 1 are all anticipated by column 9, lines 37 through 45 of Chambers. There

Chambers mentions that bridge 480 includes extensive synchronization logic, protocol conversion logic, and data buffering capabilities.

It is true Chamber indicates there is synchronization logic, protocol conversion logic, and data buffering capabilities within bridge 480; however, nowhere does Chamber disclose or suggest that this functionality is contained within separate modules. Rather, Chamber only states that logic within bridge 480 implements this functionality.

The use of four separate modules (a synchronization module, a translation module, a queue module and a driver module) is clearly set out by claim 1. This modularity has several advantages. For example, modularity of the interface block enables rapid assembly while still being tuned for a particular application. Additionally, modularity makes architectures especially suited for rapid, system-on-chip implementations.

The meaning of the term "module" is well known both in the art and in standard usage. For example, the Encarta World Dictionary (copyright 1999 by Microsoft Corporation) distributed with Microsoft Office Software (e.g., Microsoft Word 2001), defines module as "a unit that is combined with others to form a larger structure or system and is self-contained enough to be easily rearranged, replaced, or interchanged to form different structures or systems." This common definition conforms with the usage of "module" in the Specification of the present case where each module is a functional stage that can be individually configured without grossly affecting neighboring modules. See, for example, the Specification at page 4, lines 3 through 6.

No such modularity is disclosed or suggested by Chamber. Chamber merely indicates that there is synchronization logic, protocol conversion logic, and data buffering capabilities within bridge 480. Nowhere does Chamber disclose or suggest that this functionality is arranged in *modules* as the term "modules" would be ordinarily understood by persons of the art or by the public at large.

Examiner has rejected claims 4 and 6 through 12 under 35 U.S.C. § 102(e) as being anticipated by USPN 5,870,310 (Malladi). Applicant respectfully traverses the rejection.

Independent claim 4 sets out a method for providing an interface between an internal bus of an integrated circuit and a socket of a logic block within the integrated circuit.

In each of the steps of the method set out in claim 4, specific modules perform specific functions. In step (a), a synchronization module performs any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block. In step (b), a translation module is used to provide any required translation of block encoding of data transferred between the internal bus and the socket of the logic block. In step (c), a queue module is used to buffer data flowing between the internal bus and the socket of the logic block. In step (d), a driver module is used to handle low level and electrical drive specifications of the internal bus.

This use of specific modules is not disclosed or suggested by Malladi.

Particularly, Malladi does not disclose or suggest a synchronization module that performs any needed synchronization

between a clock domain of the internal bus and a clock domain of the socket of the logic block. Malladi does not disclose or suggest a translation module that is used to provide any required translation of block encoding of data transferred between the internal bus and the socket of the logic block. Malladi does not disclose or suggest a queue module that is used to buffer data flowing between the internal bus and the socket of the logic block. Malladi does not disclose or suggest a driver module that is used to handle low level and electrical drive specifications of the internal bus.

In the passages of Malladi cited by Examiner, Malladi describes certain functionality that is performed by Malladi. For example, at column 2, lines 3 through 9, Malladi states the following:

> By way of example, when various cores are designed on an IC chip, interface logic including the communication protocols, the timing requirements and the physical interconnections between cores and interface buses must be designed, laid-out, characterized, calibrated, and tested. This process is generally very costly and time consuming.

This language has been cited by Examiner as disclosing step (a) (performing any needed synchronization...) and step (b) (performing any required translation...) of claim 4. However, at column 2, lines 3 through 9, Malladi does not give any information about how the mentioned functionality is implemented. Malladi only indicates that interface logic including the communication protocols, the timing requirements and the physical interconnections between cores and interface buses must be designed, laid-out, characterized, calibrated, and tested. This language does not disclose or suggest using a synchronization *module* that performs any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block (as in step (a) of claim 4) or a translation *module* that is used to provide any required

translation of block encoding of data transferred between the internal bus and the socket of the logic block (as in step (b) of claim 4).

Since Malladi does not disclose or suggest the specific modules set out in each of steps (a) through (d) of claim 4, Malladi clearly does not anticipate the subject matter set out in claim 4.

Applicant has amended independent claim 7 to make it clear what was meant by one module performing a single function.

Independent claim 7, as amended, sets out an interface block that provides an interface between an internal bus of the integrated circuit and a socket of a logic block. The interface block comprises a plurality of modules connected in series. Each module in the pluralities performs only a single function from a plurality of functions. Any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block is a first function from the plurality of functions. Any required translation of block encoding of data is a second function from the plurality of functions. Any buffering of data flowing between the internal bus and the socket of the logic block is a third function from the plurality of functions. Any low level and electrical drive specifications of the internal bus is a fourth function from the plurality of functions. This is not disclosed or suggested by Malladi.

Examiner has suggested that data processing shell 141a is an example of an interface block. Examiner has also argued that bus interface unit (BIU) 140a, data processing core 108, and memory interface unit (MIU) 150a are modules within the interface block. However, these clearly do not function as modules in accordance with the language of claim 7.

As set out in claim 7, each module in the plurality of modules performs *only a single function* from a plurality of functions. However, bus interface unit BIU 140a is a complete interface between CPU bus 100 and data processing core 108. This means BIU 140a will provide any needed synchronization, translation, buffering and low level driving needed for data processing core 108 to access CPU bus 100. It is clear that BIU 140a performs too many functions to qualify as a module set out in claim 7 of the present application.

Likewise, memory interface unit MIU 150a is a complete interface between memory bus 102 and data processing core 108. This means MIU 150a will provide any needed synchronization, translation, buffering and low level driving needed for data processing core 108 to access memory bus 102. It is clear that MIU 150a performs too many functions to qualify as a module set out in claim 7 of the present application.

Conclusion

The use of separate modules (e.g., a synchronization module, a translation module, a queue module and a driver module) is set out in every claim of the present invention. This modular implementation of an interface block is not disclosed or suggested by the cited references.

For all the reasons discussed above, Applicant believes that this Amendment has placed the present Application in condition for allowance and favorable action is respectfully requested.

Respectfully submitted,
QUE-WON RHEE

By _____
Douglas D. Weller
Reg. No. 30,506

July 17, 2002
Santa Clara, California
(408) 985-0642

VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the Specification:

The three paragraphs found at page 5, line 20 through page 6, line 24 have been amended as follows:

--

The second stage of the configurable architecture for interface block 19 is implemented as a translation block 32. A clocked buffer 36 receives and transmits control signals from/to synchronization block ~~11~~ 31 via control lines 41 and receives and transmits data signals from/to synchronization block ~~11~~ 31 via data lines 46. Clocked buffer 36 receives and transmits control signals from/to translation block ~~12~~ 32 via control lines 51 and receives and transmits data signals from/to translation block ~~12~~ 32 via data lines 46. Translation block 32 converts the block encoding used by the virtual socket interface protocol of specialized logic block 10 to the block encoding used by the protocol implemented on on-chip bus 15. Logic within translation block 32 transforms requests used by the virtual socket interface protocol to equivalent bus requests for the protocol implemented on on-chip bus 15.

The third stage of the configurable architecture for interface block 19 is implemented as a queue block 33. A clocked buffer 37 receives and transmits control signals from/to translation block ~~12~~ 32 via control lines 42 and receives and transmits data signals from/to translation block ~~12~~ 32 via data lines 47. Clocked buffer 37 receives and transmits control signals from/to queue block 33 via control lines 52 and receives and transmits data signals from/to queue block 33 via data lines 57. Queue block 33 buffers

control signals and data signals so that information from both logic block 10 and on-chip bus 15 can flow independently.

The fourth stage of the configurable architecture for interface block 19 is implemented as a driver block 34. A clocked buffer 38 receives and transmits control signals from/to queue block 33 via control lines 43 and receives and transmits data signals from/to queue block ~~13~~ 33 via data lines 48. Clocked buffer 38 receives and transmits control signals from/to driver block 34 via control lines 53 and receives and transmits data signals from/to driver block 34 via data lines 58. Driver block 34 generates low-level electrical drive and receive specification of on-chip bus 15. Driver block 34 and on-chip bus 15 exchange control signals via control lines 44 and data signals via data lines 49.

--

In the Claims:

Claims 1 and 7 have been as follows:

1. (Amended) ~~On an integrated circuit, a~~An interface block that provides an interface between an internal bus ~~of the integrated circuit~~ and a socket of a logic block, the interface block, the internal bus and the logic block all being located within a single integrated circuit, the interface block comprising:

a synchronization module that performs any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block;

a translation module that, for data transferred between the internal bus and the socket of the logic block, provides translation of block encoding of the data;

a queue module, that buffers data flowing between the internal bus and the socket of the logic block; and,

a driver module that handles low level and electrical drive specifications of the internal bus.

7. (Amended) On an integrated circuit, an interface block that provides an interface between an internal bus of the integrated circuit and a socket of a logic block, the interface block comprising:

a plurality of modules connected in series, wherein each module in the plurality of modules performs only a single function from a plurality of functions;

~~_____~~wherein any needed synchronization between a clock domain of the internal bus and a clock domain of the socket of the logic block is a first function from the plurality of functions, any required translation of block encoding of data is a second function from the plurality of functions, any buffering of data flowing between the internal bus and the socket of the logic block is a third function from the plurality of functions, and handling any low level and electrical drive specifications of the internal bus is a fourth function from the plurality of functions ~~are performed by the plurality of modules so that one module from the plurality of modules performs a single function~~.

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☒ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☒ FADED TEXT OR DRAWING
- ☒ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☒ SKEWED/SLANTED IMAGES
- ☒ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.